ACS Statistical Machine Translation

Lecture 8: Hierarchical Phrase-based Translation

Department of Engineering
University of Cambridge

Bill Byrne

bill.byrne@eng.cam.ac.uk

Lent 2013

Statistical Machine Translation (SMT) [1]

Translate s into t:

"*Any target word sequence t is a possible translation of the input source sentence s*"

Translating $\equiv$ **Finding** the best hypothesis

$$\hat{t} = \operatorname*{argmax}_{t \in \mathcal{T}} P(t|s) = \operatorname*{argmax}_{t \in \mathcal{T}} \underbrace{P(s|t)}_{\text{Translation Model}} \underbrace{P(t)}_{\text{Language Model}}$$

- Translation Model: **from phrases to hierarchical phrases**
- Language Model is a standard N-gram model

HARD: $|\mathcal{T}|$ can be very large (at most V^I)

[1] Brown, P. et al. 1990. A Statistical Approach to Machine Translation. Computational Linguistics, Vol.16, Num.2.

Motivation

Example:

澳洲	是	与	北韩	有	邦交	的	少数	国家	之一	。
Aozhou	shi	yu	Beihan	you	bangjiao	de	shaoshu	guojia	zhiyi	.
Australia	is	with	North Korea	have	dipl. rels.	that	few	countries	one of	.

Australia is one of the few countries that have diplomatic relations with North Korea.

Limitation of Phrase-based SMT:

[Aozhou] [shi]$_1$ [yu Beihan]$_2$ [you] [bangjiao] [de shaoshu guojia zhiyi] [.]

[Australia] [has] [dipl. rels.] [with North Korea]$_2$ [is]$_1$ [one of the few countries] [.]

Distorsion limits (maximum jump distance, ...) required to avoid computational explosion prohibit the correct reordering

Motivation (2)

With **Hierarchical Phrases**:

$$\langle \text{yu } X_{\boxed{1}} \text{ you } X_{\boxed{2}}, \text{have } X_{\boxed{2}} \text{ with } X_{\boxed{1}} \rangle$$
$$\langle X_{\boxed{1}} \text{ de } X_{\boxed{2}}, \text{the } X_{\boxed{2}} \text{ that } X_{\boxed{1}} \rangle$$
$$\langle X_{\boxed{1}} \text{ zhiyi}, \text{one of } X_{\boxed{1}} \rangle$$

Translation would be possible:

[Aozhou] [shi] [[[yu [Beihan]$_1$ you [bangjiao]$_2$] de [shaoshu guojia]$_3$] zhiyi]

[Australia] [is] [one of [the [few countries]$_3$ that [have [dipl. rels.]$_2$ with [N. Korea]$_1$]]]

Hierarchical Phrase-based Translation

R_1: $S \rightarrow \langle X , X \rangle$
R_2: $S \rightarrow \langle S\ X , S\ X \rangle$
R_3: $X \rightarrow \langle s_1$, said$\rangle$
R_4: $X \rightarrow \langle s_1\ s_2$, the president said$\rangle$
R_5: $X \rightarrow \langle s_1\ s_2\ s_3$, Syrian president says$\rangle$
R_6: $X \rightarrow \langle s_2$, president$\rangle$
R_7: $X \rightarrow \langle s_3$, the Syrian$\rangle$
R_8: $X \rightarrow \langle s_4$, yesterday$\rangle$
R_9: $X \rightarrow \langle s_5$, that$\rangle$
R_{10}: $X \rightarrow \langle s_6$, would return$\rangle$
R_{11}: $X \rightarrow \langle s_6$, he would return$\rangle$

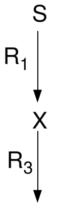
s_1	s_2	s_3	s_4	s_5	s_6
wqAl	**Alr}ys**	**Alswry**	**Ams**	**Anh**	**syEwd**

(وقال الرئيس السوري امس انه سيعود)

Hierarchical Phrase-based Translation

said

R_1: **S→⟨X , X⟩**
R_2: S→⟨S X , S X⟩
R_3: **X→⟨s_1 , said⟩**
R_4: X→⟨s_1 s_2 , the president said⟩
R_5: X→⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩

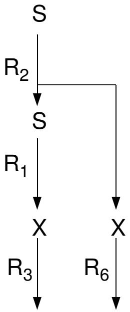


s_1	s_2	s_3	s_4	s_5	s_6
wqAl	Alr}ys	Alswry	Ams	Anh	syEwd

(وقال الرئيس السوري امس انه سيعود)

Hierarchical Phrase-based Translation

said president



s_1 s_2 s_3 s_4 s_5 s_6

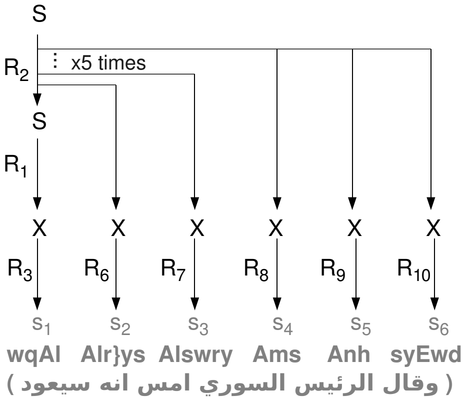
wqAl Alr}ys Alswry Ams Anh syEwd

(وقال الرئيس السوري امس انه سيعود)

R_1: S→⟨X , X⟩
R_2: **S→⟨S X , S X⟩**
R_3: X→⟨s_1 , said⟩
R_4: X→⟨s_1 s_2 , the president said⟩
R_5: X→⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: **X→⟨s_2 , president⟩**
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩

Hierarchical Phrase-based Translation

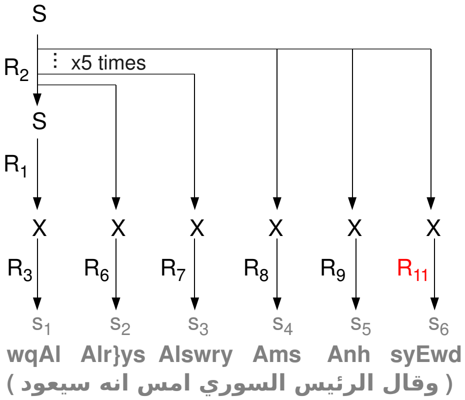
said president the Syrian yesterday that would return



R_1: $S \rightarrow \langle X , X \rangle$
R_2: $\mathbf{S \rightarrow \langle S\ X , S\ X \rangle}$
R_3: $X \rightarrow \langle s_1$, said$\rangle$
R_4: $X \rightarrow \langle s_1\ s_2$, the president said$\rangle$
R_5: $X \rightarrow \langle s_1\ s_2\ s_3$, Syrian president says$\rangle$
R_6: $X \rightarrow \langle s_2$, president$\rangle$
R_7: $\mathbf{X \rightarrow \langle s_3}$, the Syrian$\rangle$
R_8: $\mathbf{X \rightarrow \langle s_4}$, yesterday$\rangle$
R_9: $\mathbf{X \rightarrow \langle s_5}$, that$\rangle$
R_{10}: $\mathbf{X \rightarrow \langle s_6}$, would return$\rangle$
R_{11}: $X \rightarrow \langle s_6$, he would return$\rangle$

Hierarchical Phrase-based Translation

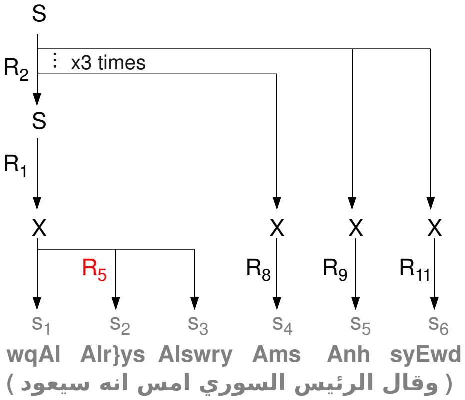
said president the Syrian yesterday that he would return



R_1: S→⟨X , X⟩
R_2: **S→⟨S X , S X⟩**
R_3: X→⟨s_1 , said⟩
R_4: X→⟨s_1 s_2 , the president said⟩
R_5: X→⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: **X→⟨s_6 , would return⟩**
R_{11}: **X→⟨s_6 , he would return⟩**

Hierarchical Phrase-based Translation

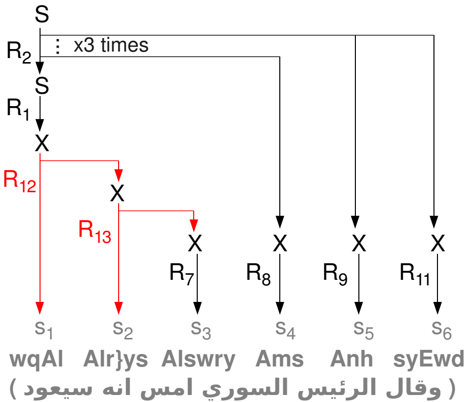
Syrian president says yesterday that he would return



R_1: S→⟨X , X⟩
R_2: S→⟨S X , S X⟩
R_3: X→⟨s_1 , said⟩
R_4: X→⟨s_1 s_2 , the president said⟩
R_5: X→⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩

Hierarchical Phrase-based Translation (2)

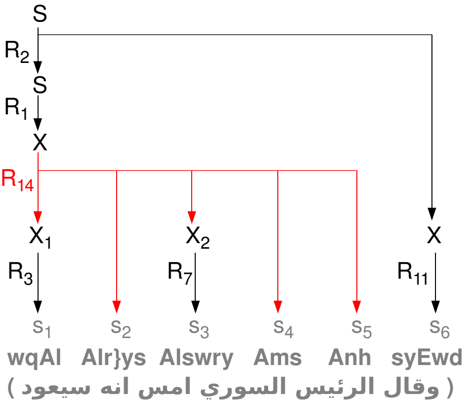
the Syrian president said yesterday that he would return



R_1: S→⟨X , X⟩
R_2: S→⟨S X , S X⟩
R_3: X→⟨s_1 , said⟩
...
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩
R_{12}: X→⟨s_1 X , X said⟩
R_{13}: X→⟨s_2 X , X president⟩

Hierarchical Phrase-based Translation (2)

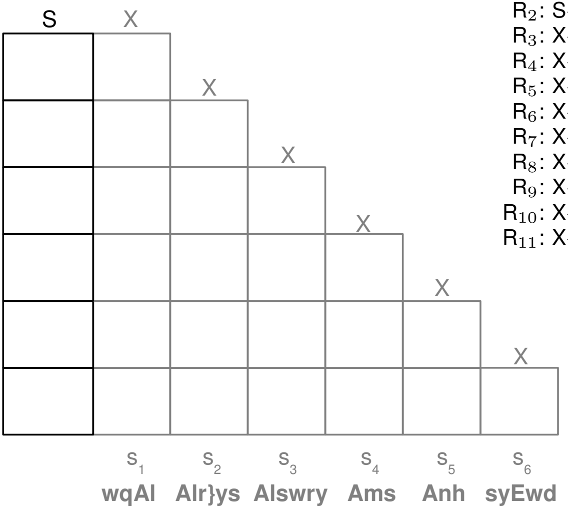
yesterday the Syrian president said that he would return



R_1: $S \rightarrow \langle X \ , \ X \rangle$
R_2: $S \rightarrow \langle S\ X \ , \ S\ X \rangle$
R_3: $X \rightarrow \langle s_1 \ , \ \text{said} \rangle$
...
R_6: $X \rightarrow \langle s_2 \ , \ \text{president} \rangle$
R_7: $X \rightarrow \langle s_3 \ , \ \text{the Syrian} \rangle$
R_8: $X \rightarrow \langle s_4 \ , \ \text{yesterday} \rangle$
R_9: $X \rightarrow \langle s_5 \ , \ \text{that} \rangle$
R_{10}: $X \rightarrow \langle s_6 \ , \ \text{would return} \rangle$
R_{11}: $X \rightarrow \langle s_6 \ , \ \text{he would return} \rangle$
R_{14}: $X \rightarrow \langle X_1\ s_2\ X_2\ s_4\ s_5 \ ,$ y'day X_2 president X_1 that$\rangle$

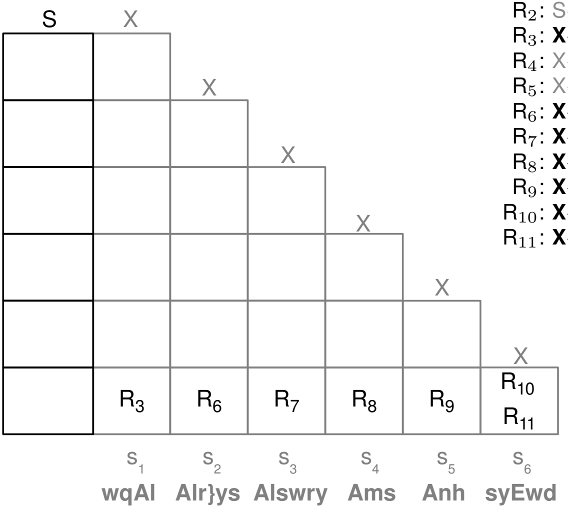
- Each rule has a probability assigned by the Translation Model

Keeping Track of All Derivations. CYK Grid



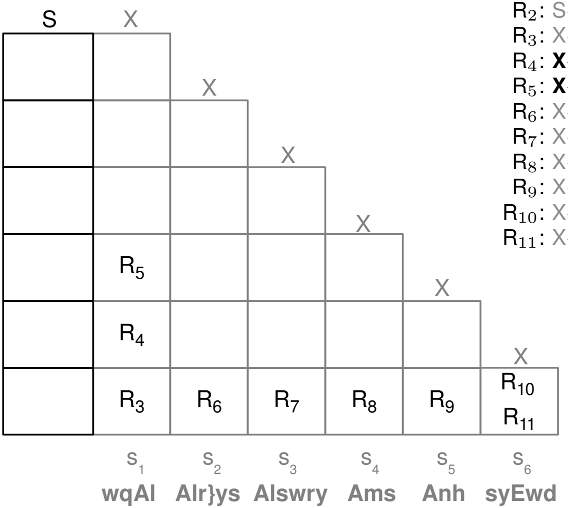
R_1: $S \rightarrow \langle X \ , \ X \rangle$
R_2: $S \rightarrow \langle S \ X \ , \ S \ X \rangle$
R_3: $X \rightarrow \langle s_1$, said$\rangle$
R_4: $X \rightarrow \langle s_1 \ s_2$, the president said$\rangle$
R_5: $X \rightarrow \langle s_1 \ s_2 \ s_3$, Syrian president says$\rangle$
R_6: $X \rightarrow \langle s_2$, president$\rangle$
R_7: $X \rightarrow \langle s_3$, the Syrian$\rangle$
R_8: $X \rightarrow \langle s_4$, yesterday$\rangle$
R_9: $X \rightarrow \langle s_5$, that$\rangle$
R_{10}: $X \rightarrow \langle s_6$, would return$\rangle$
R_{11}: $X \rightarrow \langle s_6$, he would return$\rangle$

Keeping Track of All Derivations. CYK Grid



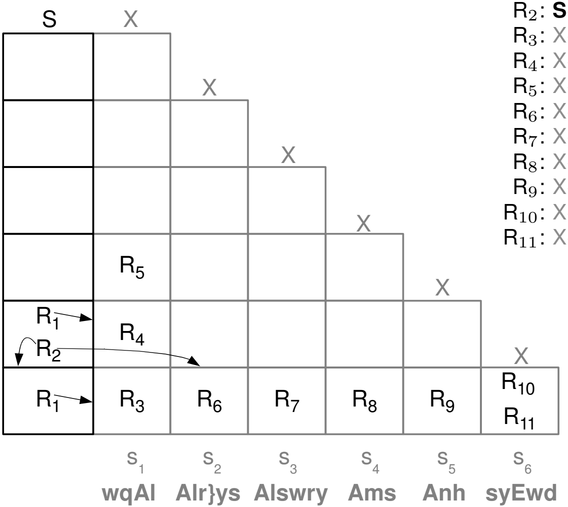
R_1: S → ⟨X , X⟩
R_2: S → ⟨S X , S X⟩
R_3: X → ⟨s_1 , said⟩
R_4: X → ⟨s_1 s_2 , the president said⟩
R_5: X → ⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: X → ⟨s_2 , president⟩
R_7: X → ⟨s_3 , the Syrian⟩
R_8: X → ⟨s_4 , yesterday⟩
R_9: X → ⟨s_5 , that⟩
R_{10}: X → ⟨s_6 , would return⟩
R_{11}: X → ⟨s_6 , he would return⟩

Keeping Track of All Derivations. CYK Grid



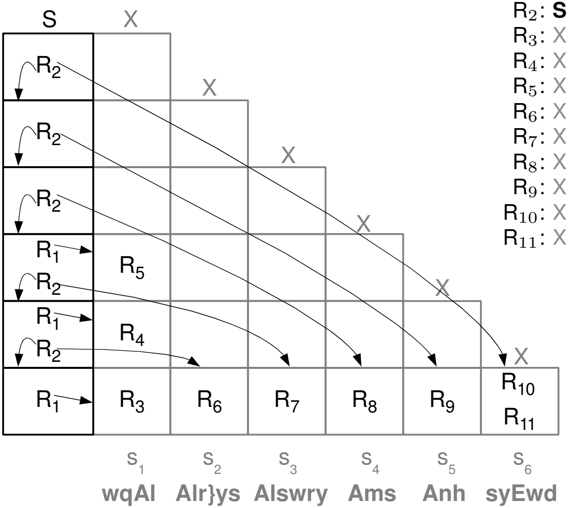
R_1: S→⟨X , X⟩
R_2: S→⟨S X , S X⟩
R_3: X→⟨s_1 , said⟩
R_4: X→⟨s_1 s_2 , the president said⟩
R_5: X→⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩

Keeping Track of All Derivations. CYK Grid



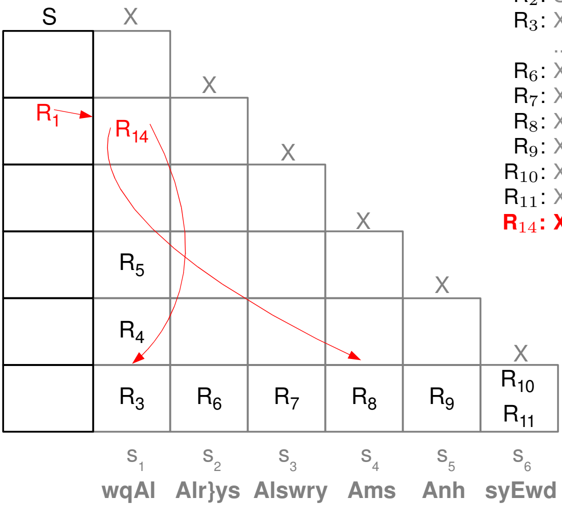
R_1: **S**$\rightarrow\langle$**X , X**$\rangle$
R_2: **S**$\rightarrow\langle$**S X , S X**$\rangle$
R_3: X$\rightarrow\langle s_1$, said$\rangle$
R_4: X$\rightarrow\langle s_1\ s_2$, the president said$\rangle$
R_5: X$\rightarrow\langle s_1\ s_2\ s_3$, Syrian president says$\rangle$
R_6: X$\rightarrow\langle s_2$, president$\rangle$
R_7: X$\rightarrow\langle s_3$, the Syrian$\rangle$
R_8: X$\rightarrow\langle s_4$, yesterday$\rangle$
R_9: X$\rightarrow\langle s_5$, that$\rangle$
R_{10}: X$\rightarrow\langle s_6$, would return$\rangle$
R_{11}: X$\rightarrow\langle s_6$, he would return$\rangle$

Keeping Track of All Derivations. CYK Grid



R_1: **S**→⟨**X , X**⟩
R_2: **S**→⟨**S X , S X**⟩
R_3: X→⟨s_1 , said⟩
R_4: X→⟨s_1 s_2 , the president said⟩
R_5: X→⟨s_1 s_2 s_3 , Syrian president says⟩
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩

Keeping Track of All Derivations. CYK Grid (2)

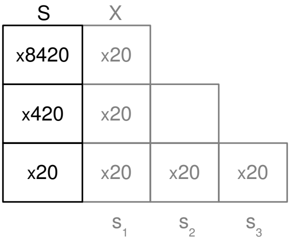


R_1: S→⟨X , X⟩
R_2: S→⟨S X , S X⟩
R_3: X→⟨s_1 , said⟩
...
R_6: X→⟨s_2 , president⟩
R_7: X→⟨s_3 , the Syrian⟩
R_8: X→⟨s_4 , yesterday⟩
R_9: X→⟨s_5 , that⟩
R_{10}: X→⟨s_6 , would return⟩
R_{11}: X→⟨s_6 , he would return⟩
R_{14}: X→⟨X_1 s_2 X_2 s_4 s_5 , y'day X_2 president X_1 that⟩

Cube Pruning Algorithm [2]

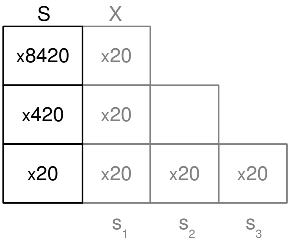
- The number of derivations can be vast
- Each derivation will produce a translation candidate
- Each candidate has a score
- Find best candidate

$$\underset{t \in \mathcal{T}}{\operatorname{argmax}} \; P(s|t) \; P(t)$$



[2]Chiang, D. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. Proc. ACL.

Cube Pruning Algorithm [2]



- The number of derivations can be vast
- Each derivation will produce a translation candidate
- Each candidate has a score
- Find best candidate

$$\underset{t \in \mathcal{T}}{\operatorname{argmax}}\ P(s|t)\ P(t)$$

- Cube-Pruning Algorithm
  - One-by-one processing of all derivations is not feasible
  - Lists of k-best hypotheses are kept in each cell (k=10^4)
  - Local decisions based on Translation and Language Model
  - ✓ Translation Model fits well in this grid representation
  - × Language Model does not: $P(t) = \prod_{n=1}^{I} p(t_n|t_{n-1})$

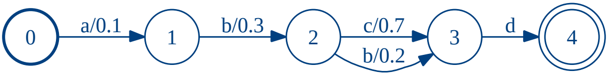
would return $\leftarrow p(return|would) \times p(would|?)$
he would return $\leftarrow p(return|would) \times p(would|he) \times p(he|?)$

- Local decisions should be avoided!

[2] Chiang, D. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. Proc. ACL.

Reviewing Weighted Finite-State Acceptors (WFSAs)

- WFSAs are devices that compactly model a formal language
- A **Weighted Acceptor** of strings ‘a b c d’ and ‘a b b d’ :



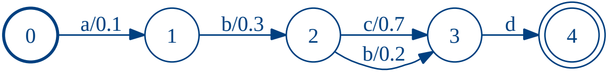
is defined by a set of states Q and a set of arcs : $q \overset{x/w}{\rightarrow} q'$

- Weighted Acceptors can assign costs to strings:
  - strings are associated with paths, which are sequences of arcs
  - weights are accumulated over paths by means of a **product operation** $\otimes$

$$w(p) = w(e_1) \ \otimes \cdots \otimes \ w(e_n)$$

Reviewing Weighted Finite-State Acceptors (WFSAs)

- WFSAs are devices that compactly model a formal language
- A **Weighted Acceptor** of strings ‘a b c d’ and ‘a b b d’ :



is defined by a set of states Q and a set of arcs : $q \stackrel{x/w}{\rightarrow} q'$

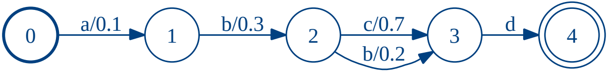
- Weighted Acceptors can assign costs to strings:
  - strings are associated with paths, which are sequences of arcs
  - weights are accumulated over paths by means of a **product operation** $\otimes$

$$w(p) = w(e_1) \otimes \cdots \otimes w(e_n)$$

Probability Semiring: $w(\text{‘a b c d’}) = 0.1 \times 0.3 \times 0.7 \times 1.0 = 0.021$ ← BEST
$w(\text{‘a b b d’}) = 0.1 \times 0.3 \times 0.2 \times 1.0 = 0.006$

Reviewing Weighted Finite-State Acceptors (WFSAs)

- WFSAs are devices that compactly model a formal language
- A **Weighted Acceptor** of strings ‘a b c d’ and ‘a b b d’ :



is defined by a set of states Q and a set of arcs : $q \stackrel{x/w}{\rightarrow} q'$

- Weighted Acceptors can assign costs to strings:
  - strings are associated with paths, which are sequences of arcs
  - weights are accumulated over paths by means of a **product operation** $\otimes$

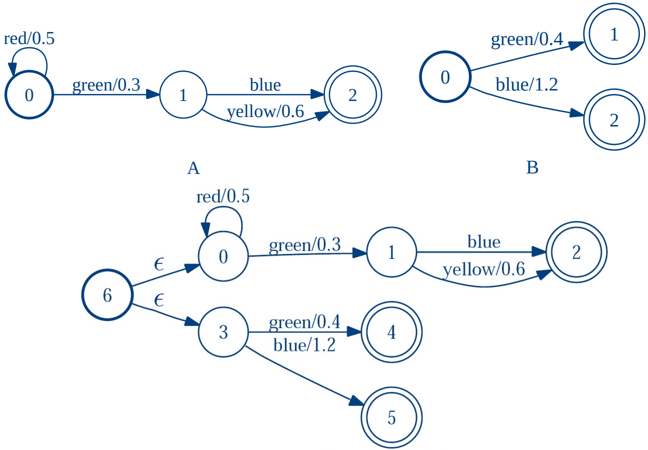
$$w(p) = w(e_1) \otimes \cdots \otimes w(e_n)$$

Tropical Semiring: $w(\text{‘a b c d’}) = 0.1 + 0.3 + 0.7 + 0.0 = 1.1$
$w(\text{‘a b b d’}) = 0.1 + 0.3 + 0.2 + 0.0 = 0.6 \leftarrow$ BEST

WFSA Operations - Union

A string x is accepted by $A = A \cup B$ if x is accepted by A or by B

$$[\![C]\!](x) = [\![A]\!](x) \bigoplus [\![B]\!](x)$$

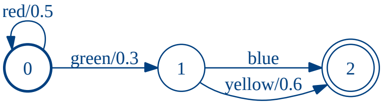


$C = A \cup B$

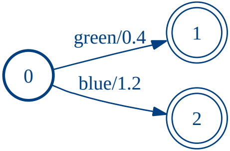
WFSA Operations - Concatenation (or Product)

A string x is accepted by $C = A \otimes B$ if x can be split into $x = x_1 x_2$ such that x_1 is accepted by A and x_2 is accepted by B

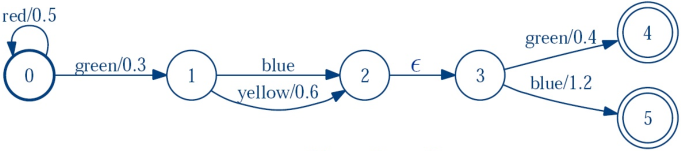
$$[\![C]\!](x) = \bigoplus_{x_1, x_2 : x = x_1 x_2} [\![A]\!](x_1) \otimes [\![B]\!](x_2)$$



A



B

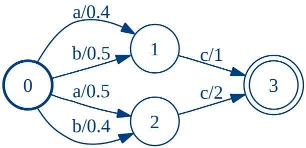


$C = A \otimes B$

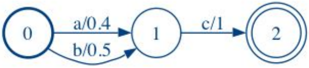
WFSA Operations for Compactness

WFSAs can be **made compact** with operations that:

- reduce their size in number of states/arcs
- accept the same distinct strings
- *the cost of each string is respected* according to the semiring

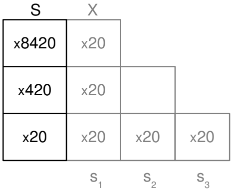


⇒



- WFSAs can represent compactly more than 10^{60} paths
- Processing a WFSA is much faster than processing all of the paths individually

HiFST. Hierarchical Translation with WFSTs [3]



- Keep all possible derivations in each cell
  Efficiently explore largest $\mathcal{T}$ in

$$\underset{t \in \mathcal{T}}{\text{argmax}}\ P(s|t)\ P(t)$$

- **Build a WFSA in each cell**
  - They compactly store millions of paths with Translation Model costs
  - We can operate with them easily and faster
  - Applying a Language Model to a WFSA is a well-established task

```
In each cell, do:

   For each rule in the cell:
      Build Rule WFSA by Concatenating target elements ( ⊗ )

   Build Cell WFSA by Unioning Rule WFSAs ( ⊕ )
```

[3] Iglesias, G. et al. 2009. Hierarchical Phrase-Based Translation with Weighted Finite State Transducers. Proc. of NAACL-HLT.

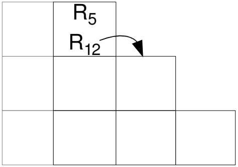
Building Rule WFSAs by Concatenation

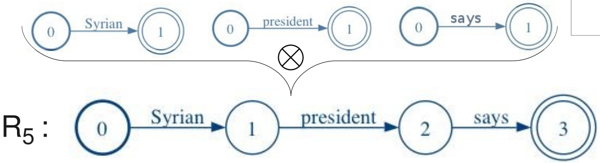
...

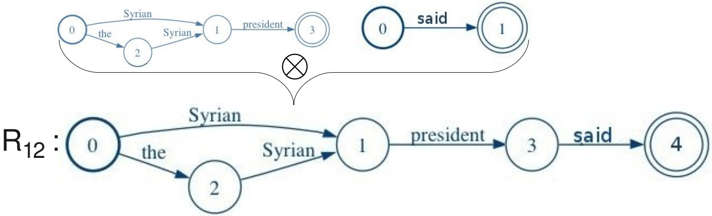
R_5: $\mathbf{X} \rightarrow \langle \mathbf{s}_1\ \mathbf{s}_2\ \mathbf{s}_3$ **, Syrian president says**$\rangle$

R_{12}: $\mathbf{X} \rightarrow \langle \mathbf{s}_1\ \mathbf{X}$ **, X said**$\rangle$

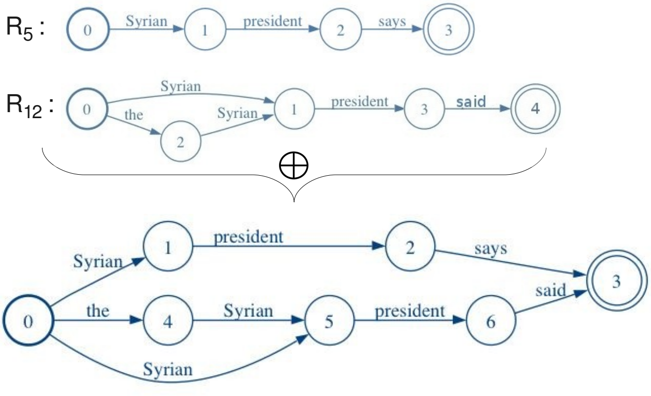
...





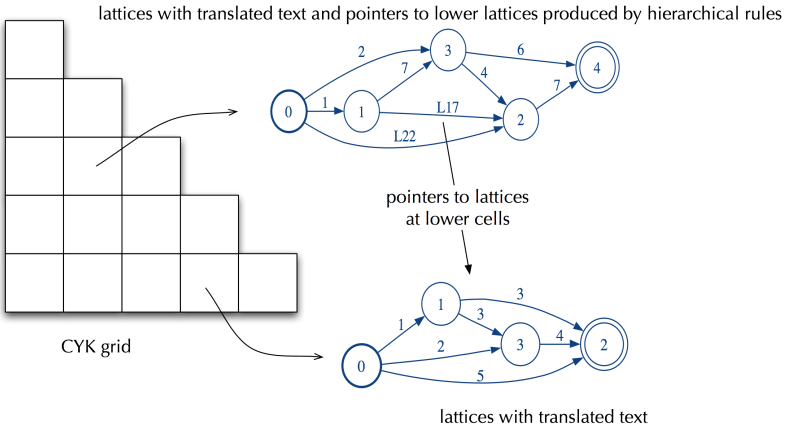


Building Cell WFSA by Union



- Can be made compact
- Target language model can be applied
- Search can be carried out efficiently

Delayed Translation



- ✓ Easy implementation with FST Replace operation
- ✓ Usual FST operations can be applied to skeleton → lattice size reduction

Pruning

Final translation lattice $L(S, 1, J)$ typically requires pruning

- Compose with target Language Model
- Perform likelihood-based pruning

Pruning in Search:

- If number of states, non-terminal category and source span meet certain conditions, then:
  - Expand Pointers in translation Lattice and Compose with Language Model
  - Perform likelihood-based pruning of the lattice
  - Remove Language Model
- Only required for certain language pairs, i.e. Chinese→English
- The hierarchical grammar can be defined to avoid this (next lecture)

Translation Experiments into English

- **Large collections of parallel text are available**
  - Arabic-to-English: ∼6M sentences, ∼150M words
  - Chinese-to-English: ∼10M sentences, ∼250M words
  - Spanish-to-English: ∼1.3M sentences, ∼37M words

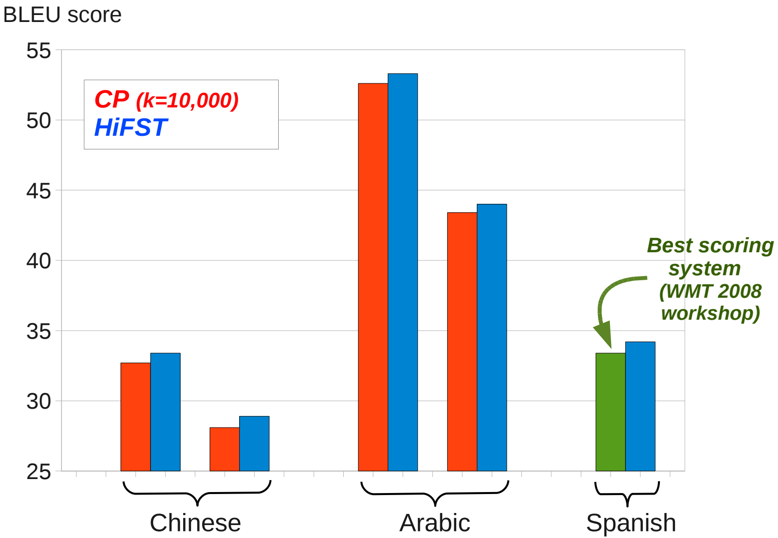
- **Hierarchical phrases are extracted from alignments**

  Maximum Likelihood estimates for $P(s|t)$

  5-gram Language Model $P(t)$

- **Contrast: Cube Pruning (CP) vs HiFST**

Translation Results into English. Contrast CP vs HiFST

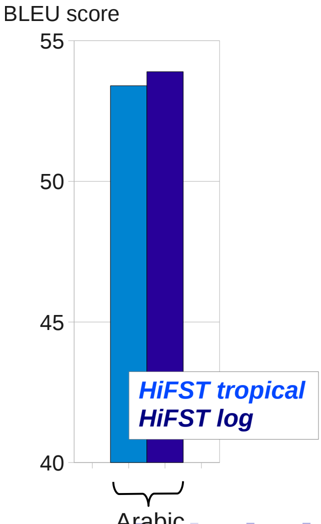


Translation Results into English. Change in Semiring

✓ Changing the Semiring is easy
Can have significant impact

- **Tropical Semiring**: Viterbi likelihood
- **Log Semiring**: Marginal probability
  i.e. sums over all derivations

✓ Additional gains with no extra programming effort



Conclusions

- ✓ **HiFST generates a bigger, richer space of translation candidates**
  Fewer Search Errors: 19% in Arabic, 48% in Chinese
  Leveraged by subsequent rescoring techniques
- ✓ **Faster decoding times**, particularly in Arabic and Spanish
- ✓ **Simple implementation**, Google OpenFST toolkit [4]
  General, well-studied algorithms
  Capable of complex semiring operations
- ✓ **HiFST system is very competititve!**
  Top-3/4 in Arabic→ and Chinese→English NIST 2012 MT Evaluation (20 participants)
  Top-1 in Arabic→English NIST 2009 MT Evaluation (22 participants)
  Top-5 in Chinese→English NIST 2008 MT Evaluation task (20 participants)
  Top-1 in Spanish→English ACL 2008 Workshop on SMT task (14 participants)

[4] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut , and M. Mohri (2007), OpenFst: A General and Efficient Weighted Finite-State Transducer Library. CIAA.